AMENDMENTS TO THE SPECIFICATION

Please amend the specification at page 7, line 23 as follows:

Generally, at a given engine rotational speed *w* (or RPM), engine unbalances may produce a vibrational response at several sensor locations on the aircraft engine and surrounding structure. Methods in accordance with the present invention may include training the neural network inverse model 112 (FIGURE 1) (*e.g.* using applied trial weights) until the model has established a mapping or correlation between applied engine unbalance and the measured responses of the engine. Since engine unbalance can vary in magnitude, angular location, and balance plane (e.g. of the fan or low pressure turbine), it is ~~preferably~~ preferable to expose the model to responses from a variety of unbalance conditions. After training, the model may then be directly applied to subsequent engines for which a balance solution is sought, preferably with no further training required. Further accuracy of the model to match specific engine-airplane installations may be obtained by repeating the training process either on-board or off-board using data gathered during flight operations each time new balance weights are added to an engine.

Please amend the specification at page 8, line 14 as follows:

Since it may be unknown how much data is needed to adequately train the neural network model, and since large amounts of test data from a single engine with many unbalance weights applied may be unavailable, an alternate approach to using experimentally-generated training data is to develop an empirical engine-airframe to generate the required responses to unbalance inputs. Such empirical models may be created using limited test cases of actual jet engine unbalance conditions and experimentally observed responses of vibration at ~~senor~~ sensor locations, vibration at instrumented component locations, and/or acoustic noise levels at various locations. Using such an empirical model approach may provide a large amount of sufficiently accurate data for the purpose of this invention because the unbalance and responses relationship in the empirical engine model can be made to match exactly at the experimentally generated test points, and then transition continuously and smoothly between the test points, as would be expected based on the physical constraints of such a system. Other known methods of empirically modeling

aircraft engines that may be suitable for evaluation of the effect of the deliberate introduction of varying degrees of system nonlinearity in a controlled manner include, for example, those methods disclosed in "Experience in Rotor Balancing of Large Commercial Jet Engines" authored by J.L. White, M.A. Heidari, and M.H. Travis, and published at SEM Proceedings of the 13th International Modal Analysis Conf., Vol. II, 1995, pp. 1338-1344, which publication is incorporated herein by reference.

Please amend the specification at page 10, line 11 as follows:

For example, FIGURE 4 is a schematic view of a process 400 process for evaluating and assessing performance of a neural network inverse model in accordance with another embodiment of the invention. In this embodiment, the process 400 includes providing vibration data at a plurality of locations on the aircraft engine, specifically, at a fan vibration sensor 402 and a low pressure turbine sensor 404. Trial engine unbalance data are also ~~are~~ provided, including trial fan unbalance data 406 and trial LPT unbalance data 408.

Please amend the specification at page 13, line 12 as follows:

In the above equations, R(n) is a diagonal N_L-by-N_L matrix, whose diagonal components are equal to or slightly less than 1. H(n) is an M-by-N_L matrix containing the partial derivatives of the output node signals with respect to the weights. P(n) is an M-by-M matrix defined as the approximate conditional error covariance matrix. A(n) is a N_L-by-N_L matrix that we refer to as the global scaling matrix. K(n) is an M-by-N_L matrix containing the Kalman gains for the weights. $\hat{W}(n)$ is a vector of length M containing ~~the~~ all the weights values. $\xi(n)$ is the error vector of the network's output layer. While the motivation for the use of artificial process noise in equation (6) was to avoid numerical difficulties, we have found in addition that it significantly enhances the performance of the GEKF algorithms in terms of rate of convergence, avoidance of local minimum and quality of solution.